GuessWhat?! Visual object discovery through multi-modal dialogue

Harm de Vries, Florian Strub, Sarath Chandar, Olivier Pietquin, Hugo Larochelle, Aaron Courville

GuessWhat?! Visual object discovery through multi-modal dialogue

Harm de Vries
University of Montreal
mail@harmdevries.com

Florian Strub
Univ. Lille, CNRS, Centrale Lille,
Inria, UMR 9189 CRIStAL
florian.strub@inria.fr

Sarath Chandar
University of Montreal
sarathcse2008@gmail.com

Olivier Pietquin
DeepMind
pietquin@google.com

Hugo Larochelle
Twitter
hlarochelle@twitter.com

Aaron Courville
University of Montreal
aaron.courville@gmail.com

Abstract

We introduce GuessWhat?!, a two-player guessing game as a testbed for research on the interplay of computer vision and dialogue systems. The goal of the game is to locate an unknown object in a rich image scene by asking a sequence of questions. Higher-level image understanding, like spatial reasoning and language grounding, is required to solve the proposed task. Our key contribution is the collection of a large-scale dataset consisting of 150K human-played games with a total of 800K visual question-answer pairs on 66K images. We explain our design decisions in collecting the dataset and introduce the oracle and questioner tasks that are associated with the two players of the game. We prototyped deep learning models to establish initial baselines of the introduced tasks.

1. Introduction

People use natural language as the most effective way to communicate, including when it comes to describe the visual world around them. They often need only a few words to refer to a specific object in a rich scene. Whenever such expressions *unambiguously* point to one object, we speak of a referring expression [23]. However, uniquely identifying the referred object is not always possible, as it depends on the listener's state of mind and the context of the scene. Many real life situations, therefore, require multiple exchanges before it is clear what object is referred to:

- Did you see that dog?
* You mean the one in the corner?
- No, the one that's running.
* Yes, what's up with that?

A computer vision system able to hold conversations about what it *sees* would be an important step towards intelligent scene understanding. Such systems would be more transparent and interpretable because humans may naturally interact with them, for example by asking clarifying questions about what it perceives. Still, a fundamental challenge remains: how to create models that understand natural language descriptions and ground them in the visual world.

Questioner	**Oracle**
Is it a vase?	Yes
Is it partially visible?	No
Is it in the left corner?	No
Is it the turquoise and purple one?	Yes

Figure 1: An example game. After a sequence of four questions, it becomes possible to locate the object (highlighted by a green bounding box).

The last few years has seen an increasing interest from the computer vision community in tasks towards this goal. Thanks to advances in training deep neural networks [16] and the availability of large-scale classification datasets [26, 35, 49], automatic object recognition has now reached human-level performance [24]. As a result, attention has been shifted toward tasks involving higher-level image understanding. One prominent example is image captioning [26], the task of automatically producing natural lan-

Is it a person?	*No*
Is it an item being worn or held?	*Yes*
Is it a snowboard?	*Yes*
Is it the red one?	*No*
Is it the one being held by the person in blue?	*Yes*

Is it a cow?	*Yes*
Is it the big cow in the middle?	*No*
Is the cow on the left?	*No*
On the right ?	*Yes*
First cow near us?	*Yes*

Figure 2: Two example games in the dataset. After a sequence of five questions we are able to locate the object (highlighted by a green mask).

guage descriptions of an image. Visual Question Answering (VQA) [6] is another popular task that involves answering single open-ended questions concerning an image. Closer to our work, the ReferIt game [21] aims to generate a single expression that refers to one object in the image.

On the other hand, there has been a renewed interest in dialogue systems [31, 37], inspired by the success of data-driven approaches in other areas of natural language processing [11]. Traditionally, dialogue systems have been built through heavy engineering and hand-crafted expert knowledge, despite machine learning attempts for almost two decades [25, 40]. One of the difficulties comes from the lack of automatic evaluation as – contrary to machine translation – there is no evaluation metric that correlates well with human evaluation [27]. A promising alternative is goal-directed dialogue tasks [31, 40, 44, 43] where agents converse to pursue a goal rather than casually chit-chat. The agent's success rate in completing the task can then be used as an automatic evaluation metric. Many tasks have recently been introduced, including the bAbI tasks [44] for testing an agent's ability to answer questions about a short story, the movie dialog dataset [12] to assess an agent's capabilities regarding personal movie recommendation and a Wizard-of-Oz framework [43] to evaluate an agent's performance for assisting users in finding restaurants.

In this paper, we bring these two fields together and propose a novel goal-directed task for multi-modal dialogue. The two-player game, called GuessWhat?!, extends the ReferIt game [21] to a dialogue setting. To succeed, both players must understand the relations between objects and how they are expressed in natural language. From a machine learning point of view, the GuessWhat?! challenge is the following: learn to acquire natural language by interaction on a visual task. Previous attempts in that direction [2, 43] do not ground natural language to their immediate environment; instead they rely on an external database through which a conversational agent searches.

The key contribution of this paper is the introduction of the GuessWhat?! dataset that contains 155,280 dialogues composed of 831,889 question/answer pairs on 66,537 images extracted from the MS COCO dataset [26]. We define three sub-tasks that are based on the GuessWhat?! dataset and prototype deep learning baselines to establish their difficulty. The paper is organized as follows. First, we explain the rules of the GuessWhat?! game in Sec. 2. Then, Sec. 3 describes how GuessWhat?! relates to previous work. In Sec. 4.1 we highlight our design decisions in collecting the dataset, while Sec. 4.2 analyses many aspects of the dataset. Sec. 5 introduces the questioner and oracle tasks and their baseline models. Finally, Sec. 6 provides a final discussion of the GuessWhat?! game.

2. GuessWhat?! game

GuessWhat?! is a cooperative two-player game in which both players see the picture of a rich visual scene with several objects. One player – the **oracle** – is randomly assigned an object (which could be a person) in the scene. This object is not known by the other player – the **questioner** – whose goal it is to locate the hidden object. To do so, the questioner can ask a series of yes-no questions which are answered by the oracle as shown in Fig 1 and 2. Note that the questioner is not aware of the list of objects, they can only see the whole picture. Once the questioner has gathered enough evidence to locate the object, they notify the oracle that they are ready to guess the object. We then reveal the list of objects, and if the questioner picks the right object, we consider the game successful. Otherwise, the game ends unsuccessfully. We also include a small penalty for every question to encourage the questioner to ask informative questions. Fig 8 and 9 in Appendix A display a full game from the perspective of the oracle and questioner, respectively.

The oracle role is a form of visual question answering where the answers are limited to *Yes*, *No* and *N/A* (not applicable). The *N/A* option is included to respond even when the question being asked is ambiguous or an answer simply cannot be determined. For instance, one cannot answer the question "*Is he wearing glasses?*" if the face of the selected person is not visible. The role of the questioner is much harder. They need to generate questions that progressively narrow down the list of possible objects. Ideally, they would like to minimize the number of questions necessary to locate the object. The optimal policy for doing so involves a binary search: eliminate half of the remaining objects with each question. Natural language is often very effective at grouping objects in an image scene. Such strategies depend on the picture, but we distinguish the following types:

Spatial reasoning We group objects spatially within the image scene. One may use absolute spatial informa-

tion – *Is it on the bottom left of the picture?* – or relative spatial location – *Is it to the left of the blue car?*.

Visual properties We group objects by their size – *Is it big?*, shape – *Is it square?* – or color – *Is it blue?*.

Object taxonomy We can use the hierarchical structure of object categories, i.e. taxonomy, to group objects e.g. *Is it a vehicle?* to refer to both cars and trucks.

Interaction We group objects by how we interact with them – *Can you drive it?*.

The goal of the GuessWhat?! task is to enable machines to understand natural descriptions and ground them into the visual world. Note that such higher-level reasoning only occurs when the scene is rich enough i.e. when there are enough objects in the scene. People otherwise tend to fall back to a linear search strategy by simply enumerating objects (often by their category names).

3. Related work

The GuessWhat?! game and the data collected from it present opportunities for the extension of current research on image captioning, visual question answering and dialogue systems. In the following, we describe previous work in these areas and relate them to the open challenges offered by GuessWhat?!. We also mention other relevant work on dataset collection.

Image captioning Our work builds on top of the MS COCO dataset [26] which consists of 120k images with more than 800k object segmentations. In addition, the dataset provides 5 captions per image which initiated an explosion of interest from the research community into generating natural language descriptions of images. Several methods have been proposed [20, 42, 45], all inspired by the encoder-decoder approach [11, 41] that has proven successful for machine translation. Image captioning research uncovered successful approaches to automatically generate coherent, factual statements about images. Modeling the interactions in GuessWhat?! requires instead to model the process of asking useful questions about images.

VQA datasets Visual Question Answering (VQA) tasks form another well known extension of the captioning task. They instead require answering a question given a picture (*e.g. "How many zebras are there in the picture?", "Is it raining outside?"*). Recently, the VQA challenge [6] has provided a new dataset far bigger than previous attempts [15, 29] where, much like in GuessWhat?!, questions are free-form. An extensive body of work has followed from this publication, largely building on the image captioning literature [3, 28, 39, 46]. Unfortunately, many of these advanced methods were shown to marginally improve on simple baselines [19]. Recent work [3] also reports that trained models often report the same answer to a question irrespective of the image, suggesting that they largely exploit predictive correlations between questions and answers present in the dataset. The GuessWhat?! game and dataset attempt to circumvent these issues. Because of the questioner's aim to locate the hidden object, the generated questions are different in nature: they naturally favour spatial understanding of the scene and the attributes of the objects within it, making it more valuable to consult the image. Besides, it only contains binary questions, whose answers we find to be balanced and has twice more questions on average per picture.

Goal-directed dialogue GuessWhat?! is also relevant to the goal-directed dialogue research community. Such systems are aimed at collaboratively achieving a goal with a user, such as retrieving information or solving a problem. Although goal-directed dialogue systems are appealing, they remain hard to design. Thus, they are usually restricted to specific domains such as train ticket sales, tourist information or call routing [32, 40, 47]. Besides, existing dialogue datasets are either limited to fewer than 100k example dialogues [12], unless they are generated with template formats [12, 43, 44] or simulation [33, 36] in which case they don't reflect the free-form of natural conversations. Finally, recent work on end-to-end dialogue systems fail to handle dynamical contexts. For instance, [43] intersects a dialogue with an external database to recommend restaurants. Well-known game-based dialogue systems [1, 2] also rely on static databases. In contrast, GuessWhat?! dialogues are heavily grounded by the images. The resulting dialogue is highly contextual and must be based on the content of the current picture rather than an external database. Thus, to the best of our knowledge, the GuessWhat?! dataset marks an important step for dialogue research, as it is the first large scale dataset that is both goal-oriented and multi-modal.

Human computation games GuessWhat?! is in line with Von Ahn's seminal work on human computation games [4, 5] who showed that games are an effective way to gather labeled data. The first ESP game [4] was developed to collect image tags, and was later extended to Peekaboom [5] to gather object segmentations. These games were developed more than a decade ago, when object recognition was in its infancy and served a different purpose than GuessWhat?!.

ReferIt Probably closest to our work is the ReferIt game [21, 30, 48]. In this game, one player observes an annotated object in a scene, for which they need to generate an expression that refers to it (*e.g. ïthe man wearing the white t-shirt¨*). The other player then receives this expression and subsequently clicks on the location of the object within the image. The original dataset [21] uses the IMAGEClef dataset [13], while three recent extensions [30, 48] were built on top of MS COCO. All three databases select images with only $2-4$ objects of the *same* category. In contrast,

GuessWhat?! picks images with $3-20$ objects without further restrictions on the object class, and thus contains three times more images than the ReferIt datasets. To further investigate the difference between ReferIt and GuessWhat?!, we compare three samples for the *same* selected object in Fig 14 in Appendix B. While ReferIt directly locates the object with a single expression, GuessWhat?! iteratively narrows down the object by means of positive and *negative* feedback on questions. We also observe that GuessWhat?! dialogues favor more abstract concepts, such as "*Is it edible?*" or "*Is it on oval plate?*" than ReferIt.

4. GuessWhat?! Dataset

4.1. Data collection

Images We use a subset of the training and validation images and objects of the MS COCO dataset [26]. We first discard objects that are too small (area $< 500\text{px}^2$) to be decently located by a human observer. Then, we only keep images containing three to twenty objects, to avoid trivial or overly complicated images. In total, we keep 77,973 images with 609,543 objects. We verified that this selection does not significantly alter the original dataset distribution.

Amazon Mechanical Turk The data collection was crowd-sourced on Amazon Mechanical Turk (AMT) [9]. We created two separate tasks – known as HITs on AMT – for the questioner and oracle roles, and rewarded the questioner slightly more than the oracle. We ensured the quality of the data collection by several means. First, the workers had to go through a qualification round which consisted of successfully completing 10 games while producing fewer than 4 mistakes or disconnects. After qualification, HITs continue to consist of a batch of 10 successful games. We incentivize the worker to produce as many successful dialogues in a row by providing bonuses for making fewer mistakes. Secondly, players could report on each other and players were banned after a certain number of reports. Thus, players were incentivized to cooperate. In the end, we only kept dialogues from qualified people and successful dialogues from the qualification round. In contrast to traditional dataset collection, our game requires an interactive session between two players. Fortunately, we found that the GuessWhat?! game was highly engaging. A total of more than 10K people participated in our HITs, and our top ten participants played over $2,000$ games each. Since questions were manually typed, they could contain spelling mistakes. Thus, we retrieved all questions containing words that do not occur in an English dictionary and manually corrected the 1000 most common words. For the remaining 30k questions, we created two HITs that to correct the spelling mistakes. See Figure 10 in Appendix A for further details.

	Full	Finished	Success
# dialogues	155,280	144,434	131,394
# questions	821,889	732,081	648,493
# words	3,986,192	3,540,497	3,125,219
# voc. size	11,465	10,985	10,469
# voc. size (3+)	5,444	5,179	4,919
# images	66,537	65,112	62,954
# objects	134,073	125,349	114,271

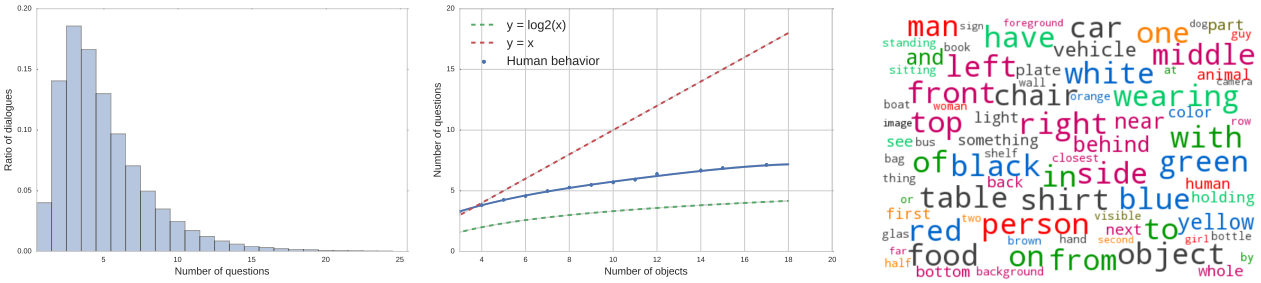
Table 1: GuessWhat?! statistics split by dataset types.

4.2. Data analysis

In the following, we explore properties of the data we collected using the GuessWhat?! game. We provide global statistics, examine the vocabulary used by the questioners and highlight the relationship between properties of objects to guess and the odds of having a successful dialogue.

Dataset statistics The raw GuessWhat?! dataset is composed of 155,280 dialogues containing 821,889 question/answer pairs on 66,537 unique images and 134,073 unique objects. The answers are respectively 52.2% *no*, 45.6% *yes* and 2.2% *N/A*. On average, there are 5.2 questions per dialogue and 2.3 dialogues per image. The dialogues contain 3,986,192 word tokens in total, making up 11,465 different words with at least one occurrence and 5,444 words with at least 3 occurrences. Moreover, 84.6% of the dialogues are successful, 8.4% are unsuccessful and 7.0% are not completed (disconnection, timeout etc.). Thus, different subsets co-exist in the GuessWhat?! dataset, we will refer to the dataset as full, finished and successful when we include all the dialogues, all finished dialogues (successful and unsuccessful) or only successful dialogues, respectively. For more details, the previous statistics are broken down into dataset types in Tab 1.

Question distributions To get a better understanding of the GuessWhat?! games, we show the number of questions within a dialogue and the average number of questions given the number of objects within a image in Fig 3. First, the number of questions within a dialogue decreases exponentially, as players tend to shorten their dialogues to speed up the game (and therefore maximize their gains). More interestingly, we observe that the average number of questions given the number of objects within an image appears to follow a function that grows at a rate between logarithmically and linearly. A questioning strategy of simply listing objects (e.g. "*is it the chair*", etc.) would imply linear growth in the number of questions, while the optimal binary search strategy would imply logarithmic growth. Thus the human questioners seem to imply a strategy that is somewhere in between. We conjecture three reasons why humans do not achieve the optimal search strategy. First, the questioner does not have access to the ground truth list of objects in the picture, and might, therefore, overestimate the number of objects. Second, some humans tend to favor a linear search



(a) (b) (c)

Figure 3: (a) Number of questions per dialogue (b) Number of questions per dialogue vs the number of objects within the picture (c) Word cloud of GuessWhat?! vocabulary with each word proportional to its frequency. Words are colored based on a hand-crafted clustering. Uninformative words such as *"it"*, *"is"* are manually removed.

strategy. Finally, the questioner may ask additional questions to confirm that he has located the right object. This can be important in the presence of possible oracle errors.

Vocabulary To gain insight into the vocabulary used by the questioner, we compute the frequency of words in the GuessWhat?! corpus and display the most frequent words as a word cloud in Fig 3c. Several key words clearly stand out. As explained in Sec. 2, some of those key words refer to abstract object properties such as *person* or *object*, spatial locations such as *right/left* or *side* and visual features such as *red/black/white*. Furthermore, prepositions are also heavily used to express relationships between objects. To better understand the sequential aspect of the questions, we study the evolution of the vocabulary at each question round. We observe that questioners use abstract object properties such as *human/object/furniture* only at the beginning of the dialogues, and quickly switch to either spatial or visual terms such as *left/right*, *white/red* or *table,chair*. This can be highlighted by applying a Dynamic Topic Model [8] to study the evolution of topics over the course of the dialogue as shown in Fig 19 in Appendix C.

Elements of success To investigate whether certain object properties favour success, we compute the success ratio of dialogues relative to: the size of the unknown objects in Fig 4b, the number of objects within the images in Fig 4a, the object category, the location of objects within the images and the size of the dialogues in Fig 20, Fig 21a and Fig 21b in Appendix C, respectively. As one may expect, the more complex the scene is, the lower the success rate is. When there are only 3 objects, the questioner has 95% success rate, while this ratio drops to around 70% with 20 objects. Similarly, big objects are almost always found while the smallest one are only found 60% of the time. Questioners easily find objects in the middle of the picture but have more difficulties to find them on the border. Finally, objects from categories that are often grouped together, e.g. *bananas* or *books*, have a lower success rates.

Miscellaneous In Fig 4c we break down the ratio of yes-no answers within the dialogues. While the first yes-no answers are balanced for small dialogues, they often terminate with a final *yes*. In contrast, long dialogues often start with a higher proportion of negative answers which slowly decrease during the exchange.

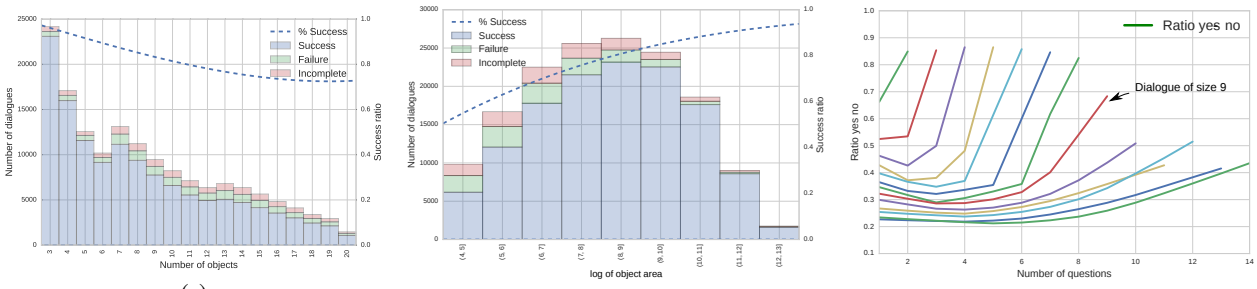
4.3. Dataset release

We split the GuessWhat?! dataset by randomly assigning 70%, 15% and 15% of the *images* and its corresponding dialogues to the training, validation and test set. This way of dividing the data ensures that we evaluate performance on images not seen during training. The GuessWhat?! dataset is available at `https://guesswhat.ai/download`.

5. Baselines

We now empirically investigate the difficulty of the oracle and questioner tasks. To do so, we trained reasonable baselines for each task and measured their performance.

Formally, a GuessWhat?! game revolves around an image $I \in \mathbb{R}^{M \times N}$ containing a set of K segmented objects $\{O_1, \ldots, O_K\}$. Each object O_k is assigned an object category $c_k \in \{1, \ldots, C\}$ and has a pixel-wise segmentation mask $S_k \in \{0, 1\}^{M \times N}$ to specify its location and size. The game further consists of a sequence of questions and answers $D = \{q_1, a_1, \ldots, q_J, a_J\}$, produced by the questioner and oracle. We will use $q_{<j}$ and $a_{<j}$ to refer to the first $j-1$ questions and answers, respectively. Each question q_j contains a sequence of N_j tokens, i.e. $q_j = \{w_{j1}, \ldots, w_{jN_j}\}$, where w_{ji} is taken from a vocabulary V and represents the token at position i in question j. Each answer is either *Yes*, *No* or *N/A*, i.e. $a_j \in \{\text{Yes, No, N/A}\}$. Finally, the oracle has access to the identity of the correct object O_{correct}, and the prediction of the questioner will be denoted as O_{predict}.



(a) (b) (c)

Figure 4: (a-b) Histogram of absolute/relative successful dialogues with respect to the number of objects and the size of the objects, respectively. (c) Evolution of answer distribution clustered by the dialogue length

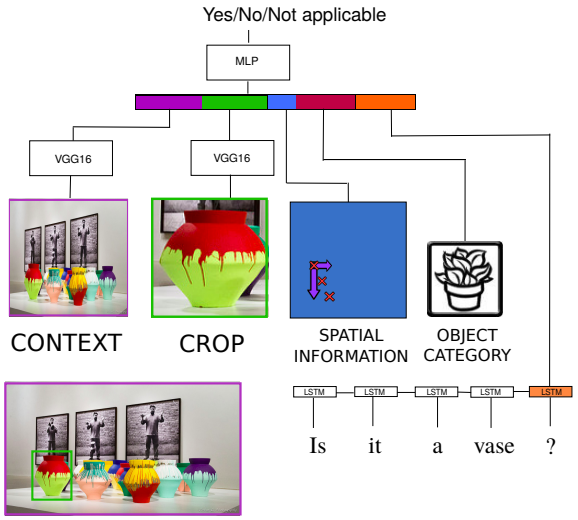


Figure 5: An schematic overview of the "Image + Question + Crop + Spatial + Category" oracle model.

5.1. Oracle baselines

The oracle task requires to produce a yes-no answer for any object within a picture given a natural language question. We first introduce our model and then outline its results to get a better understanding of the GuessWhat?! dataset.

Model We propose a simple neural network based approach to this model, illustrated in Fig 5. Specifically, we use an appropriate neural network architecture to embed each of the following information: the image I, the cropped object from S, its spatial information, its category c and the current question q. These embeddings are then concatenated as a single vector and fed as input to a single hidden layer MLP that outputs the final answer distribution using a softmax layer. Finally, we minimize the cross-entropy error during the training and report the classification error at evaluation time.

The details on how we compute the embeddings are as follows. To embed the full image, it is rescaled to a 224 by 224 image and is passed through a pre-trained VGG network to obtain its FC8 features. As for the selected object, it is first cropped by finding the smallest rectangle that encapsulates it, based on its segmentation mask. We then rescale the crop to a 224 by 224 square, before obtaining its FC8 features from the pre-trained VGG network. Although we could use the mask to drop out pixels around the selected object, we keep the crop as is since pre-trained VGG networks are exposed to such background noise during their training. We also embed the spatial information of the crop, to help locate the cropped object within the whole image. To do so, we follow the approach of [18, 48] and extract an 8-dimensional vector of the location of the bounding box:

$$x_{spatial} = [x_{min}, y_{min}, x_{max}, y_{max}, x_{center}, y_{center}, w_{box}, h_{box}] \quad (1)$$

where w_{box} and h_{box} denote the width and height of the bounding box, respectively. We normalize the image height and width such that coordinates range from -1 to 1, and place the origin at the center of the image. As for the object category, we convert its one-hot class vector into a dense category embedding using a learned look-up table. Finally, the embedding of the current natural language question q is computed using an Long Short-Term Memory (LSTM) network [17] where questions are first tokenized by using the word punct tokenizer from the python nltk toolkit [7]. For simplicity, we decided to ignore the question-answer pairs history $q_{<t}$ in our oracle baseline.

Training setting We train all oracle models on the full dataset. During training, we keep the parameters of the VGG network fixed, and optimize the LSTM, object category/word look-up tables and MLP parameters by minimizing the negative log-likelihood of the correct answer. We use ADAM [22] for optimization and train for at most 15 epochs. We use early stopping on the validation set, and report the train, valid and test error.

Results We report results for several oracle models using a different set of inputs in Table 2. We name the

Model	Train err	Val err	Test err
Dominant class (no)	47.4%	46.2%	50.9%
Question	40.2%	41.7%	41.2%
Image	45.7%	46.7%	46.7%
Crop	40.9%	42.7%	43.0%
Question + Crop	22.3%	29.1%	29.2%
Question + Image	37.9%	40.2%	39.8%
Question + Category	23.1%	25.8%	25.7%
Question + Spatial	28.0%	31.2%	31.3%
Question + Category + Spatial	17.2%	21.1%	**21.5%**
Question + Category + Crop	20.4%	24.4%	24.7%
Question + Spatial + Crop	19.4%	26.0%	26.2%
Question + Category + Spatial + Crop	16.1%	21.7%	22.1%
Question + Spatial + Crop + Image	20.7%	27.7%	27.9%
Question + Category + Spatial + Image	19.2%	23.2%	23.5%

Table 2: Classification errors for the oracle baselines on train, valid and test set. The best performing model is "Question + Category + Spatial" and refers to the MLP that takes the question, the selected object class and its spatial features as input.

model after the input we feed to it. For instance, (Question+Category+Spatial+Image) refers to the network fed with the question q, the object category c, the spatial features $x_{spatial}$ and the full image I. The results of all subsets are reported in Table 6 in Appendix C.

Because the GuessWhat?! dataset is fairly balanced, simply outputting the most common answer in the training set – No – results in a high 50.8% error rate. Solely providing the image or crop features barely improves upon this result. Only using the question slightly improves the error rate to 41.2%. We speculate that this small bias comes from questioners that refer to objects that are never segmented or overrepresented categories. As hoped, we observe that the error rate significantly drops ($< 31\%$) when we finally feed information on the object to guess (crop, spatial or category) to the model. We find that crop and category information are redundant: the (Question+Category) and (Question+Crop) model achieve respectively 29.2% and 25.7% error, while the combined model (Question+Category+Crop) achieves 24.7%. In general, we expect the object crop to contain additional information, such as color information, beside the object class. However, we find that the object category outperforms the object crop embedding. This might be partly due to the imperfect feature extraction from the crops. Finally, our best performing model combines object category and its spatial features along with the question.

5.2. Questioner baselines

Given an image, the questioner must ask a series of questions and guess the correct object. We separate the questioner task into two different sub-tasks that are trained independently:

Guesser Given an image I and a sequence of questions and answers D_J, predict the correct object O_{correct} from the set of all objects O.

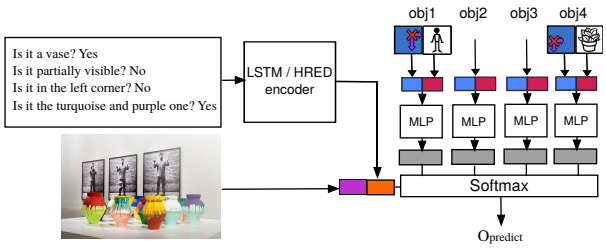


Figure 6: Overview of the guesser model for an image with 4 segmented objects. The weights are shared among the MLPs, this allows for an arbitrary number of objects.

Model	Train err	Val err	Test err
Human	9.0%	9.2%	9.2%
Random	82.9%	82.9%	82.9%
LSTM	27.9%	37.9%	**38.7%**
HRED	32.6%	38.2%	39.0%
LSTM+VGG	26.1%	38.5%	39.5%
HRED+VGG	27.4%	38.4%	39.6%

Table 3: Classification errors for the guesser baselines on train, valid and test set.

Question Generator Given an image I and a sequence of T questions and answers $D_{\leq T}$, produce a new question q_{T+1}.

In general, one also needs a module to determine when to start guessing the object (and stop asking questions). In our baseline, we bypass this issue by fixing the number of questions to 5 for the question generator model.

Guesser The role of the guesser model is to predict the correct object. To do so, the guesser has access to the image, the dialogue and the list of objects in the image. We encode the image by extracting its FC8 features from VGG16 network. A dialogue of a GuessWhat?! game is a sequence on two different levels: there is a variable number of question-answer pairs where each question in turn consists of a variable-length sequence of tokens. This can be encoded into a fixed size vector by using either an LSTM encoder [17] or an HRED encoder [38]. While the LSTM encoder considers the dialogue as one flat sequence, HRED explicitly models the hierarchy by two different Recurrent Neural Networks (RNN). First, an encoder RNN creates a fixed-size representation of a question or answer by reading in its tokens and taking the last hidden state of the RNN. This representation is then processed by the context RNN to obtain a representation of the current dialogue *state*. For both models, we concatenate the image and dialogue features and do a dot-product with the embedding for all the objects in the image, followed by a softmax to obtain a prediction distribution over the objects. Given the best performance of the "Question+Category+Spat" oracle model, we

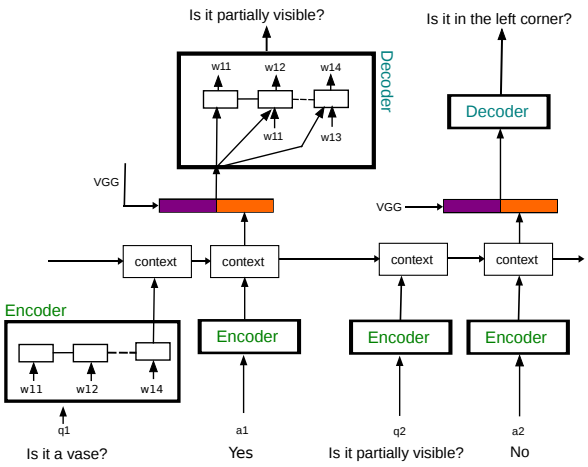


Figure 7: HRED model conditioned on the VGG features of the image. To avoid clutter, we here only show the part of the model that defines a distribution over the third question given the first two questions, its answers and the image $P(q_2|q_{<2}, a_{<2}, I)$. The complete HRED model models the distribution over all questions.

represent objects by their category and their spatial features. More precisely, we concatenate the 8-dimensional spatial representation (see Eq. 1) and the object category look-up and pass it through an MLP layer to get an embedding for the object. Note that the MLP parameters are shared to handle the variable number of objects in the image. See Fig 6 for an overview of the guesser with HRED and LSTM.

Table 3 reports the results for the guesser baselines using human-generated dialogues. As a first baseline, we report the performance of a random guesser which does not use the dialogue information. We split the guesser results based on whether they use the VGG features or not. In general, we find that including VGG features does not improve the performance of the LSTM and HRED models. We hypothesize that the VGG features are a too coarse representation of the image scene, and that most of the visual information is already encoded in the question and the object features. Surprisingly, we find LSTMs to perform slightly better than the sophisticated HRED models.

Question Generator The question generation task is hard for several reasons. First, it requires high-level visual understanding to ask meaningful questions. Second, the generator should be able to handle long-term context to ask a sequence of relevant questions, which is one of the most challenging problems in dialogue systems. Additionally, we evaluate the question generator using the imperfect oracle and imperfect guesser, which introduces compounding errors.

Hierarchical recurrent encoder decoder (HRED) [38] is the current state of the art method for natural language generation tasks. We extend this model by conditioning on the

Model	Error
Human generated dialogue	38.7%
QGen+GT	53.2%
QGen+ORACLE	66.0%
Random	82.9%

Table 4: Test error for the question generator models (QGEN) based on VGG+HRED(FT) guesser model. We here report the accuracy error of the guesser model fed with the questions from the QGEN model.

VGG features of the image as illustrated in Fig 7. Finally, we train our proposed model by maximizing the conditional log-likelihood:

$$\log P(Q|A, I) = \log \prod_{j=1}^{J} P(q_j|q_{<j}, a_{<j}, I) \quad (2)$$

$$= \log \prod_{j=1}^{J} \prod_{i=1}^{N_j} P(w_{ji}|w_{j<i}, a_{\leq j}, I) \quad (3)$$

with respect to the described parameters. At test time, we use a beam-search to approximately find the most probable question q_j. Evaluating the questioner model requires a pre-trained oracle and a pre-trained guesser model. We use our questioner model to first generate a question which is then answered by the oracle model. We repeat this procedure 5 times to obtain a dialogue. We then use the best performing guesser model to predict the object and report its error as the metric for the QGEN model. Since we use ground truth answers during the QGEN training while we use oracle answers at test time, there is a mismatch between the training and testing procedure. This can be avoided by using the oracle answers also during training time. We call these models QGEN+GT and QGEN+ORACLE respectively.

Table 4 shows the results. A guesser based on human generated dialogues achieves 38.7% error. The Question Generator models achieve reasonable performance which lies in between the random performance and the performance of the guesser on human dialogues. We observe that using the Oracle's answers while training the Question Generator introduces additional errors which significantly deteriorates performance. Some example dialogues generated by the QGen+GT model are shown in Fig. 22 and 23.

6. Discussion

We introduced the GuessWhat?! game, a novel framework for multi-modal dialogue. To the best of our knowledge, we present the first large-scale dataset involving images and dialogue. A wide range of challenges may arise from this union as they rely on different fields of machine learning such as natural language understanding, generative models or computer vision. GuessWhat?! turns out to be an engaging game that greatly decreases the cost for collec-

tion of a big dataset required for modern algorithms. As a second contribution, we introduced three tasks based on the questioner and oracle role. In each case, we prototyped a neural architecture as a first baseline. We analyzed these results and presented a quantitative description of the GuessWhat?! dataset.

We believe GuessWhat?! could allow for a myriad of other applications that may either be based on the game itself or extending the database to other tasks. For instance, it can be interesting to compute a confidence interval before proceeding to the final guess. Differently, GuessWhat?! could be a test bed for one-shot learning [14] of guessing new object categories, transfer learning on line-drawing images [10] or using questions from another language. Thus, the GuessWhat?! dataset offers an opportunity to develop original machine learning tasks upon it.

Acknowledgement The authors would like to acknowledge the stimulating environment provided by the MILA and SequeL labs. We thank all members of the MILA lab who participated in a trial run of the data collection, and all workers of AMT who participated in our HITs. We thank Jake Snell, Mengye Ren, Laurent Dinh, Jeremie Mary and Bilal Piot for helpful discussions. We acknowledge the following agencies for research funding and computing support: NSERC, Calcul Québec, Compute Canada, the Canada Research Chairs and CIFAR, CHISTERA IGLU and CPER Nord-Pas de Calais/FEDER DATA Advanced data science and technologies 2015-2020. SC is supported by a FQRNT-PBEEE scholarship.

References

[1] 20 Questions. http://www.20q.net/. Accessed: 2016-09. 3

[2] Akinator. en.akinator.com/. Accessed: 2016-09. 2, 3

[3] A. Agrawal, D. Batra, and D. Parikh. Analyzing the Behavior of Visual Question Answering Models. *arXiv preprint arXiv:1606.07356*, 2016. 3

[4] L. V. Ahn and L. Dabbish. Labeling images with a computer game. In *Proc. of the SIGCHI conference on Human factors in computing systems*. ACM, 2004. 3

[5] L. V. Ahn, R. Liu, and M. Blum. Peekaboom: a game for locating objects in images. In *Proc. of the SIGCHI conference on Human Factors in computing systems*. ACM, 2006. 3

[6] S. Antol, A. Agrawal, J. Lu, M. Mitchell, D. Batra, Z. Lawrence, and D. Parikh. Vqa: Visual question answering. In *Proc. of ICCV*, 2015. 2, 3, 18

[7] S. Bird, E. Klein, and E. Loper. *Natural language processing with Python*. O'Reilly Media, Inc., 2009. 6

[8] D. Blei and J. Lafferty. Dynamic topic models. In *Proc. ICML*, 2006. 5, 18

[9] M. Buhrmester, T. Kwang, and S. Gosling. Amazon's Mechanical Turk a new source of inexpensive, yet high-quality, data? *Perspectives on psychological science*, 6(1):3–5, 2011. 4

[10] L. Castrejon, Y. Aytar, C. Vondrick, H. Pirsiavash, and A. Torralba. Learning Aligned Cross-Modal Representations from Weakly Aligned Data. In *Proc. CVPR*, 2016. 9

[11] K. Cho, B. V. Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio. Learning phrase representations using RNN encoder-decoder for statistical machine translation. In *Proc. of EMNLP*. Association for Computational Linguistics, 2014. 2, 3

[12] J. Dodge, A. Gane, X. Zhang, A. Bordes, S. Chopra, A. Miller, A. Szlam, and J. Weston. Evaluating prerequisite qualities for learning end-to-end dialog systems. In *Proc. of ICLR*, 2016. 2, 3

[13] H. Escalante, C. Hernández, J. Gonzalez, A. López-López, M. Montes, E. Morales, E. Sucar, L. Villaseñ, and M. Grubinger. The segmented and annotated IAPR TC-12 benchmark. *CVIU*, 2010. 3

[14] L. Fei-Fei, R. Fergus, and P. Perona. One-shot learning of object categories. *IEEE transactions on pattern analysis and machine intelligence*, 2006. 9

[15] D. Geman, S. Geman, N. Hallonquist, and L. Younes. Visual turing test for computer vision systems. *Proceedings of the National Academy of Sciences*, 112(12):3618–3623, 2015. 3

[16] I. Goodfellow, Y. Bengio, and A. Courville. Deep Learning. Book in preparation for MIT Press, 2016. 1

[17] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997. 6, 7

[18] R. Hu, H. Xu, M. Rohrbach, J. Feng, K. Saenko, and T. Darrell. Natural Language Object Retrieval. *Proc. of CVPR*, 2016. 6

[19] A. Jabri, A. Joulin, and L. van der Maaten. Revisiting Visual Question Answering Baselines. In *Proc of ECCV*, 2016. 3

[20] A. Karpathy and L. Fei-Fei. Deep visual-semantic alignments for generating image descriptions. In *Proc. CVPR*, 2015. 3

[21] S. Kazemzadeh, V. Ordonez, M. Matten, and T. Berg. ReferItGame: Referring to Objects in Photographs of Natural Scenes. In *Proc. of EMNLP*, 2014. 2, 3

[22] D. P. Kingma and J. Ba. Adam: A Method for Stochastic Optimization. *CoRR*, abs/1412.6980, 2014. 6

[23] E. Krahmer and K. V. Deemter. Computational generation of referring expressions: A survey. *Computational Linguistics*, 38(1):173–218, 2012. 1

[24] Y. LeCun, Y. Bengio, and G. Hinton. Deep learning. *Nature*, 521(7553):436–444, 2015. 1

[25] E. Levin and R. Pieraccini. A stochastic model of computer-human interaction for learning dialogue strategies. In *Eurospeech*, volume 97, pages 1883–1886, 1997. 2

[26] T. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and L. Zitnick. Microsoft coco: Common objects in context. In *Proc of ECCV*, 2014. 1, 2, 3, 4

[27] C. Liu, R. Lowe, I. Serban, M. Noseworthy, L. Charlin, and J. Pineau. How NOT to evaluate your dialogue system: An empirical study of unsupervised evaluation metrics for dialogue response generation. *arXiv preprint arXiv:1603.08023*, 2016. 2

[28] J. Lu, J. Yang, D. Batra, and D. Parikh. Hierarchical Question-Image Co-Attention for Visual Question Answering. *arXiv preprint arXiv:1606.00061*, 2016. 3

[29] M. Malinowski and M. Fritz. A multi-world approach to question answering about real-world scenes based on uncertain input. In *Proc. of NIPS*, pages 1682–1690, 2014. 3

[30] J. Mao, J. Huang, A. Toshev, O. Camburu, A. Yuille, and K. Murphy. Generation and comprehension of unambiguous object descriptions. *arXiv preprint arXiv:1511.02283*, 2015. 3

[31] O. Lemon and O. Pietquin, editor. *Data-Driven Methods for Adaptive Spoken Dialogue Systems*. Springer, 2012. 2

[32] O. Pietquin and T. Dutoit. A probabilistic framework for dialog simulation and optimal strategy learning. *IEEE Transactions on Audio, Speech, and Language Processing*, 2006. 3

[33] O. Pietquin and H. Hastie. A survey on metrics for the evaluation of user simulations. *The knowledge engineering review*, 28(01):59–73, 2013. 3

[34] R. Řehůřek and P. Sojka. Software Framework for Topic Modelling with Large Corpora. In *Proc. LREC 2010 Workshop on New Challenges for NLP Frameworks*, 2010. 18

[35] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, et al. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3):211–252, 2015. 1

[36] J. Schatzmann, K. Weilhammer, M. Stuttle, and S. Young. A survey of statistical user simulation techniques for reinforcement-learning of dialogue management strategies. *The knowledge engineering review*, 21(02):97–126, 2006. 3

[37] I. Serban, R. Lowe, L. Charlin, and J. Pineau. A survey of available corpora for building data-driven dialogue systems. *arXiv preprint arXiv:1512.05742*, 2015. 2

[38] I. Serban, A. Sordoni, Y. Bengio, A. Courville, and J. Pineau. Hierarchical neural network generative models for movie dialogues. *arXiv preprint arXiv:1507.04808*, 2015. 7, 8

[39] K. Shih, S. Singh, and D. Hoiem. Where to look: Focus regions for visual question answering. In *Proc. of CVPR*, 2016. 3

[40] S. Singh, M. Kearns, D. Litman, and M. Walker. Reinforcement Learning for Spoken Dialogue Systems. In *Proc. of NIPS*, 1999. 2, 3

[41] I. Sutskever, O. Vinyals, and Q. Le. Sequence to sequence learning with neural networks. In *Proc of NIPS*, 2014. 3

[42] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan. Show and tell: A neural image caption generator. In *Proc. of CVPR*, 2015. 3

[43] T. Wen, M. Gasic, N. Mrksic, L. Rojas-Barahona, P. Su, S. Ultes, D. Vandyke, and S. Young. A Network-based End-to-End Trainable Task-oriented Dialogue System. *arXiv preprint arXiv:1604.04562*, 2016. 2, 3

[44] J. Weston, A. Bordes, S. Chopra, A. Rush, B. van Merriënboer, A. Joulin, and T. Mikolov. Towards ai-complete question answering: A set of prerequisite toy tasks. In *Proc. of ICLR*, 2016. 2, 3

[45] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhutdinov, R. Zemel, and Y. Bengio. Show, attend and tell: Neural image caption generation with visual attention. 2015. 3

[46] Z. Yang, X. He, J. Gao, L. Deng, and A. Smola. Stacked attention networks for image question answering. In *Proc. of CVPR*, 2016. 3

[47] S. Young, M. Gašić, B. Thomson, and J. Williams. POMDP-based statistical spoken dialog systems: A review. *Proc. of the IEEE*, 101(5):1160–1179, 2013. 3

[48] L. Yu, P. Poirson, S. Yang, A. Berg, and T. Berg. Modeling context in referring expressions. In *Proc. in ECCV*. Springer, 2016. 3, 6

[49] B. Zhou, A. Lapedriza, J. Xiao, A. Torralba, and A. Oliva. Learning deep features for scene recognition using places database. In *Proc of NIPS*, 2014. 1

A. User interface

Figure 8, 9 presents the instructions for the oracle and questioner before they started their first game.

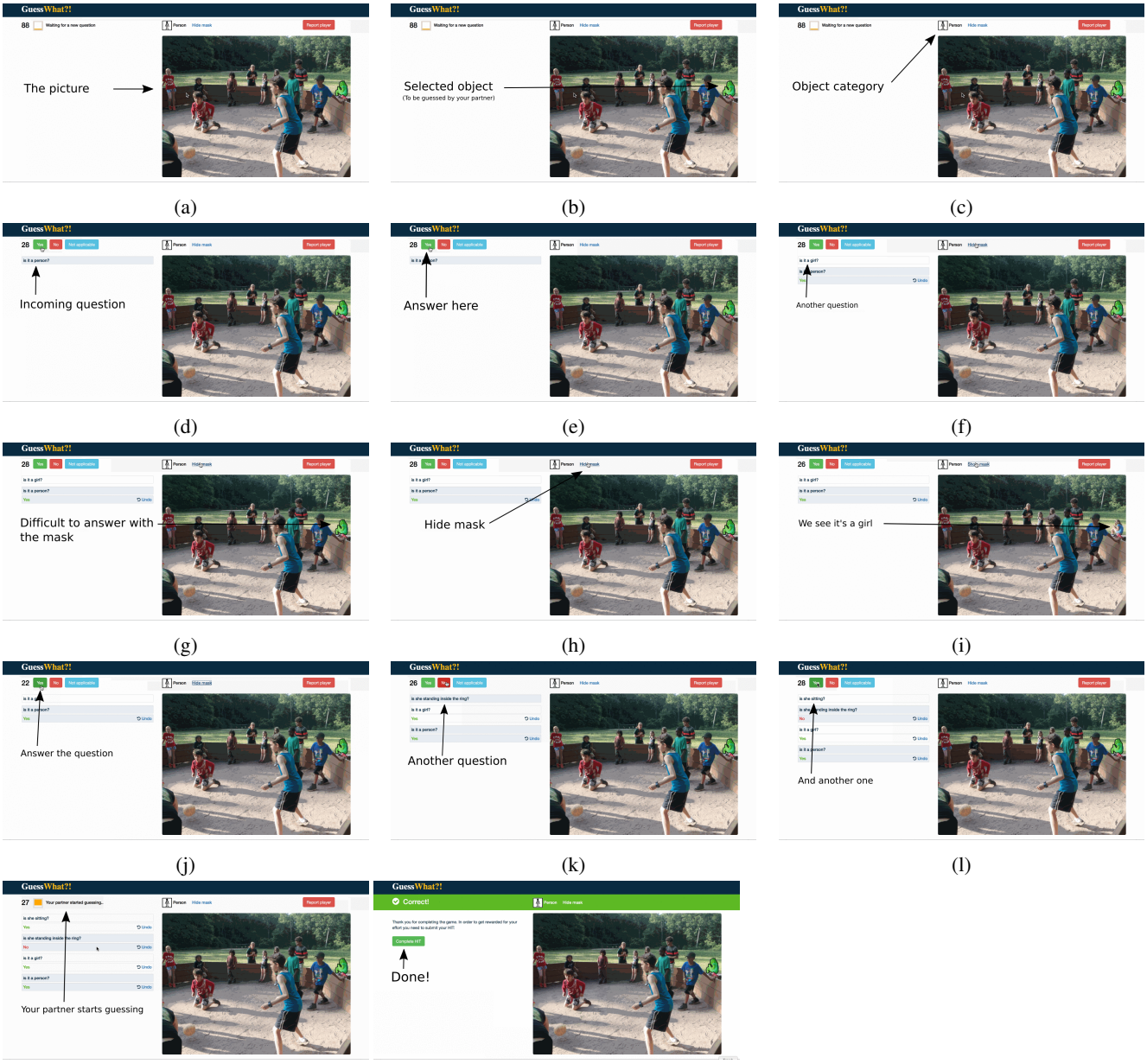


Figure 8: An example game from the perspective of the oracle. Shown from left to right and top to bottom.

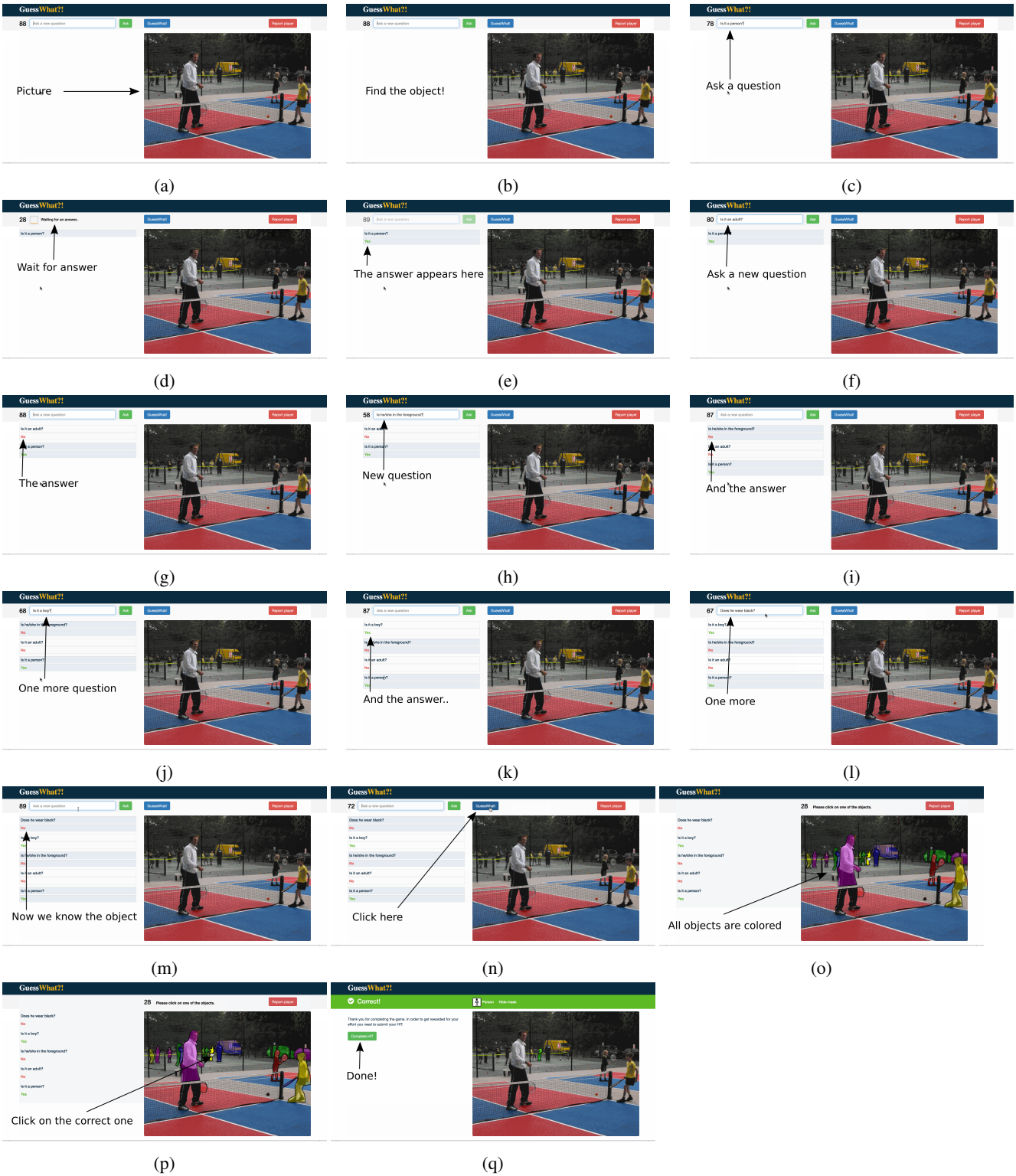
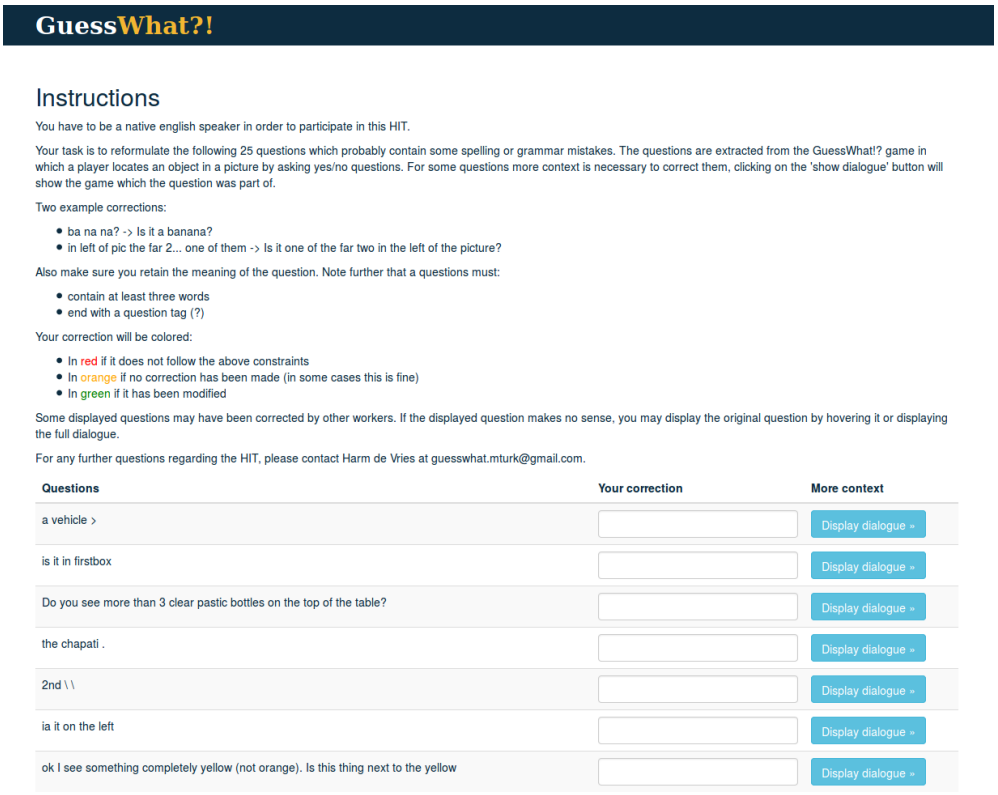
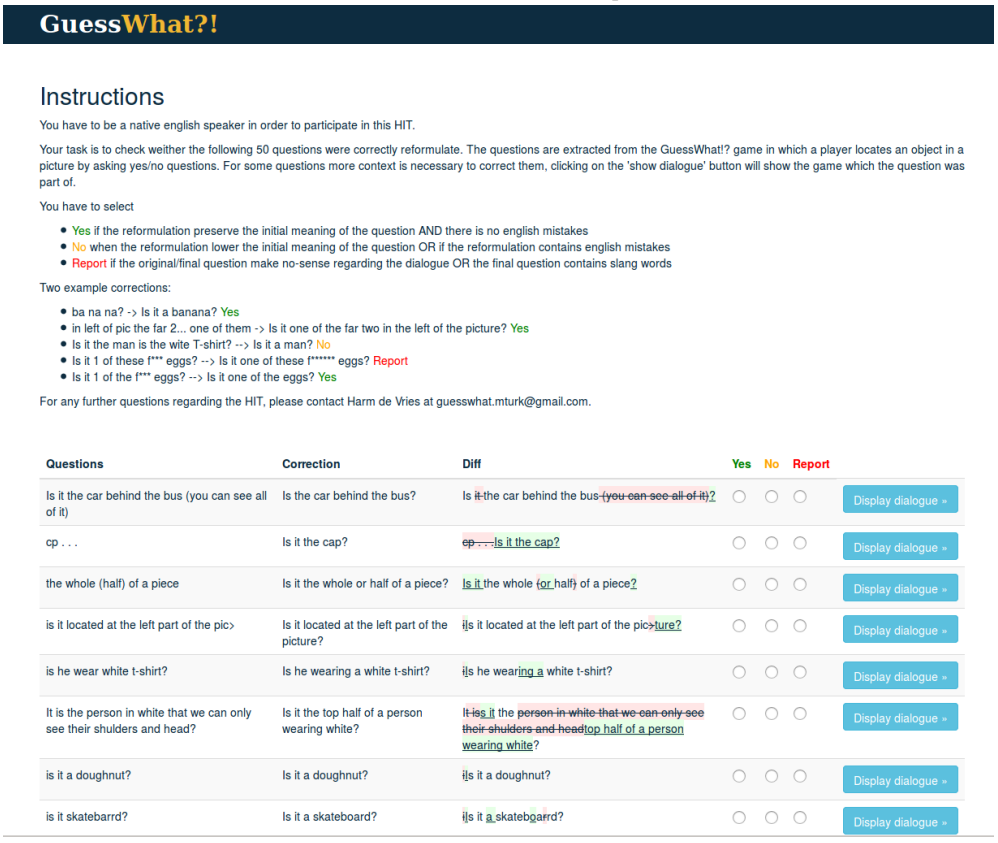


Figure 9: An example game from the perspective of the questioner. Shown from left to right and top to bottom.



(a) Interface to fix ill-formatted questions



(b) Interface to validate the fix ill-formatted questions

Figure 10: In the first task, we ask workers to correct mistakes in the questions. We then ask workers to validate the proposed correction by showing the difference between the original question and its correction. We alternate both tasks till all questions are corrected and validated.

B. GuessWhat?! samples

Is it a person?	*No*
Is it a cat?	*Yes*
Is it in the hands of these girls?	*No*
The cat in the right side of the image?	*Yes*

Is it human?	*Yes*
Are they five humans visible?	*Yes*
Is it leftmost?	*No*
Is it in the middle?	*Yes*
Is it the third one from the left?	*Yes*

Is it an ear?	*No*
Is it a horn?	*No*
Is it a cow?	*Yes*
Is it the one at the back ?	*No*
Does it have a horn?	*Yes*

Figure 11: Three other examples of our dataset.

Is it a person?	*Yes*
Is it in the foreground?	*No*
Is he wearing blue shirt?	*No*
Is he wearing black shirt?	*Yes*

Is it a person?	*Yes*
Is it standing?	*Yes*
Is it in yellow?	*Yes*

Is it a person?	*Yes*
One in yellow?	*No*
One with white pants?	*No*
One with blue pants in the top?	*No*

Figure 12: Same picture but different objects.

Is it a person?	*No*
Is it on the shelves?	*No*
Is it on the floor?	*Yes*
Is it a cup?	*No*
Is it blue?	*No*
Is it wood?	*No*
Is it the bed?	*No*
Is it in the lower half of the image?	*Yes*
Is it in the lower left corner?	*No*
Is it near the blue tray with two cups?	*No*
Is it near the green and blue toy?	*No*
Is it near the boy laying on the floor?	*Yes*
Is it something he is touching?	*Yes*
Is it the remote?	*Yes*
Is it the one in his right hand (close to the wooden box)?	*No*

Figure 13: A long dialogue example in a very rich environment.

ReferIt

woman in red jacket with green bag

left woman in red coat

GuessWhat?!

Is it a person?	Yes
One of the people with the stroller on the right?	No
One of the two people crossing the street towards us?	No
The woman in red?	Yes

ReferIt

guy with hat bottom right front

guy sitting with hat bottom right

GuessWhat?!

Is it a person?	Yes
Are they standing?	No
Are they touching the frisbee	No
Are they holding a square thing?	Yes
Black cap ?	Yes

ReferIt

doughnut in the middle with green frosting

GuessWhat?!

Is it edible?	Yes
Is it on oval plate?	Yes
Is it green?	Yes
The whole doughnut?	Yes

Figure 14: Samples illustrating the difference between GuessWhat?! and ReferIt games. As both dataset are constructed on top of MS COCO, we picked identical objects (and images).

C. Additional database statistics

Figure 15 presents a word co-occurrence matrix of the GuessWhat?! dataset. Figure 16 and Figure 17a compares the object size and category distribution of GuessWhat?! with MS Coco.

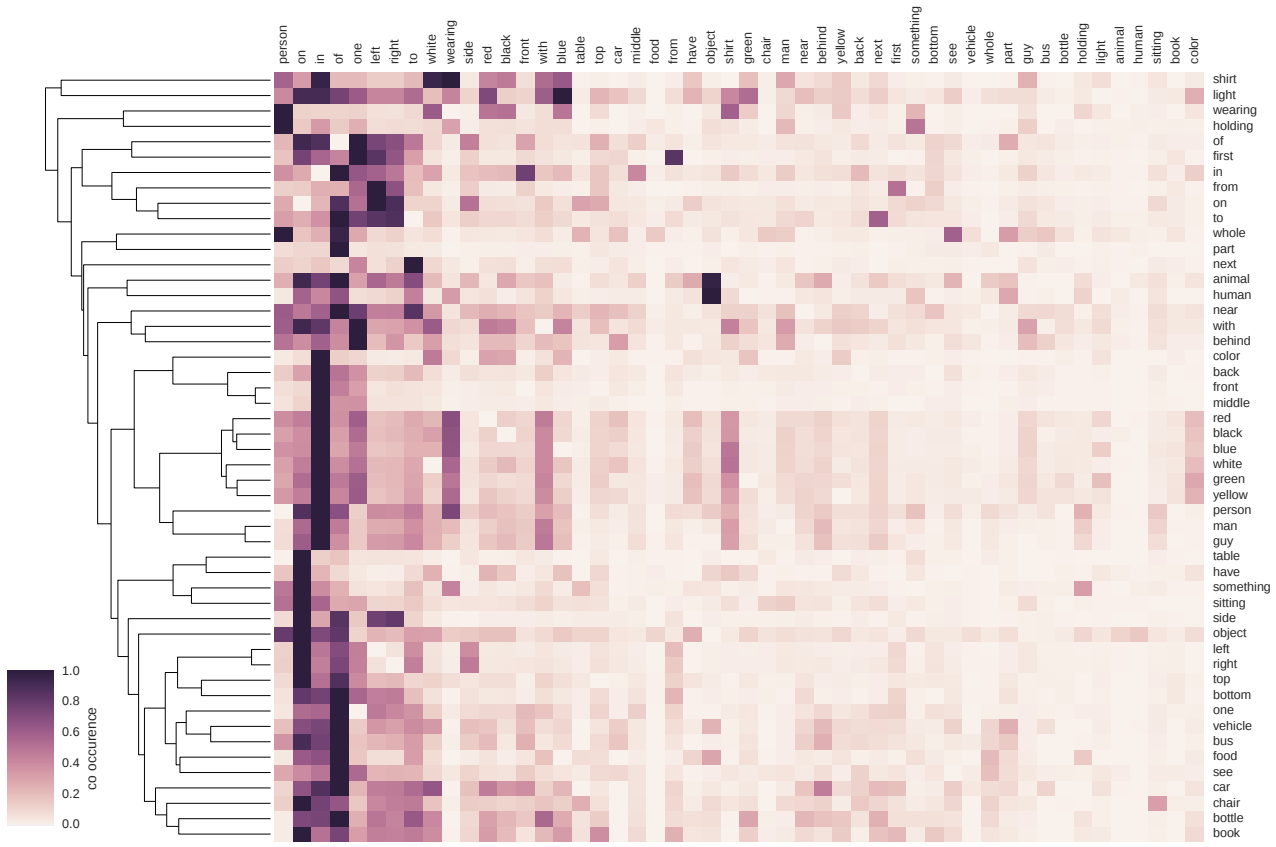


Figure 15: Co-occurrence matrix of words. Only the 50 most frequent words are kept. Rows are first normalized before being sorted thanks a hierarchical clustering with an euclidean distance.

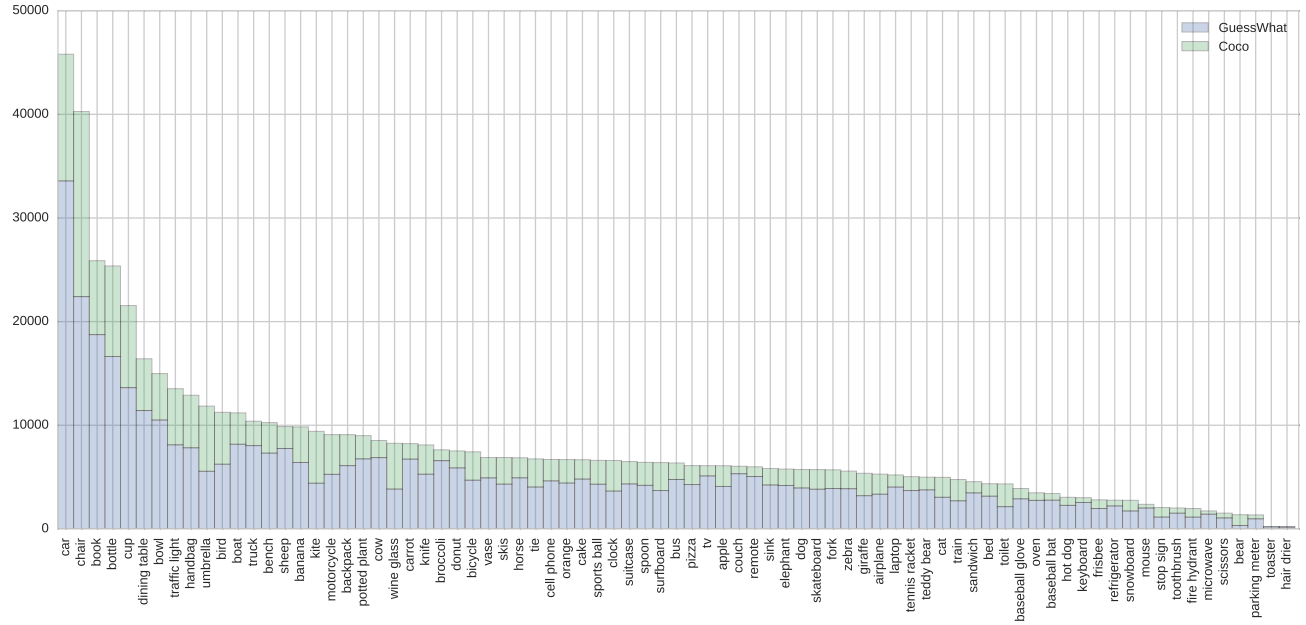
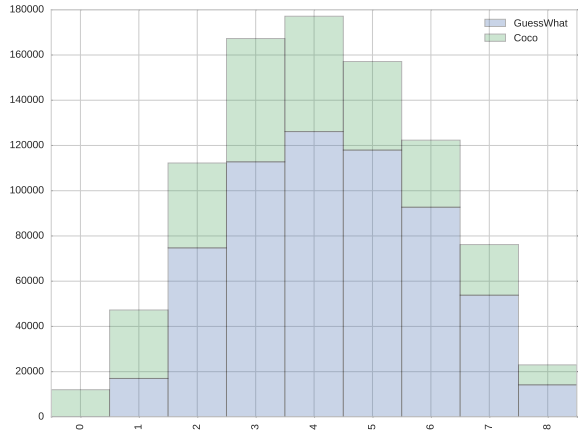
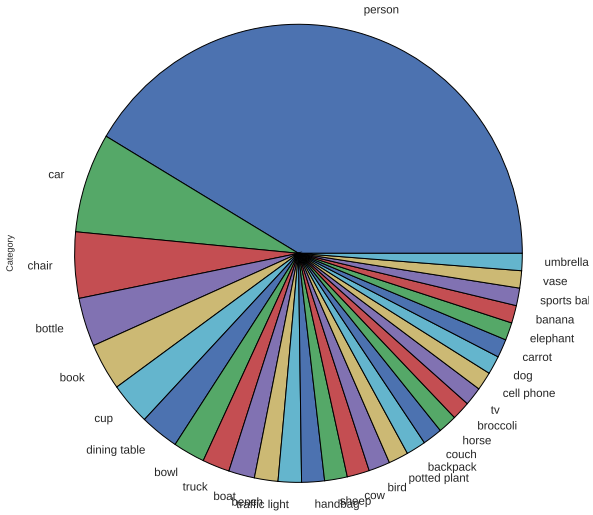


Figure 16: Visualization of the object category distribution of MS COCO and GuessWhat?! dataset. The person category was removed for clarity (resp. 273469 and 188204).

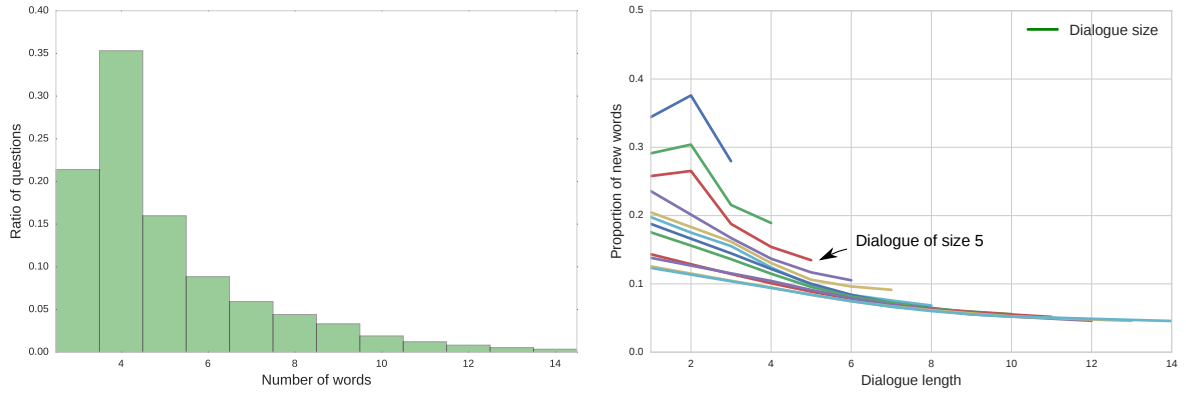


(a)



(b)

Figure 17: (a) Visualization of the object size distribution of MS COCO and GuessWhat?! dataset. (b) Distribution of the the 30 (out of 80) prominent object categories in the GuessWhat?! which represent 71.3% of the objects.



(a) (b)

Figure 18: (a) Number of words per question. The question length follows a Poisson-like distribution, a finding which is in line with other datasets [6]. (b) Percentage of apparition of new words along a dialogues. Questioner tends keep using the same words during the dialogues.

Topic 1	Topic 2
Abstract words	Descriptive words
person	left
food	one
vehicle	right
human	wearing
car	side
one	white
object	red
animal	table

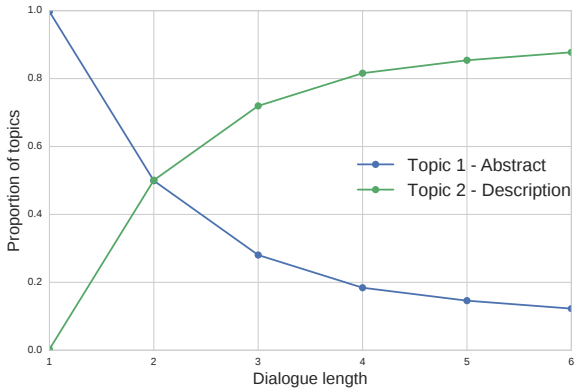


Figure 19: Relative evolution of topics during a dialogue of size 6. We applied Data Topic Models (DTM) [8] with the python framework [34] on our dataset. The table reports the two prominent detected topics with their respective key words while the figure display their relative evolution during the dialogue. The topic titles are manually picked.

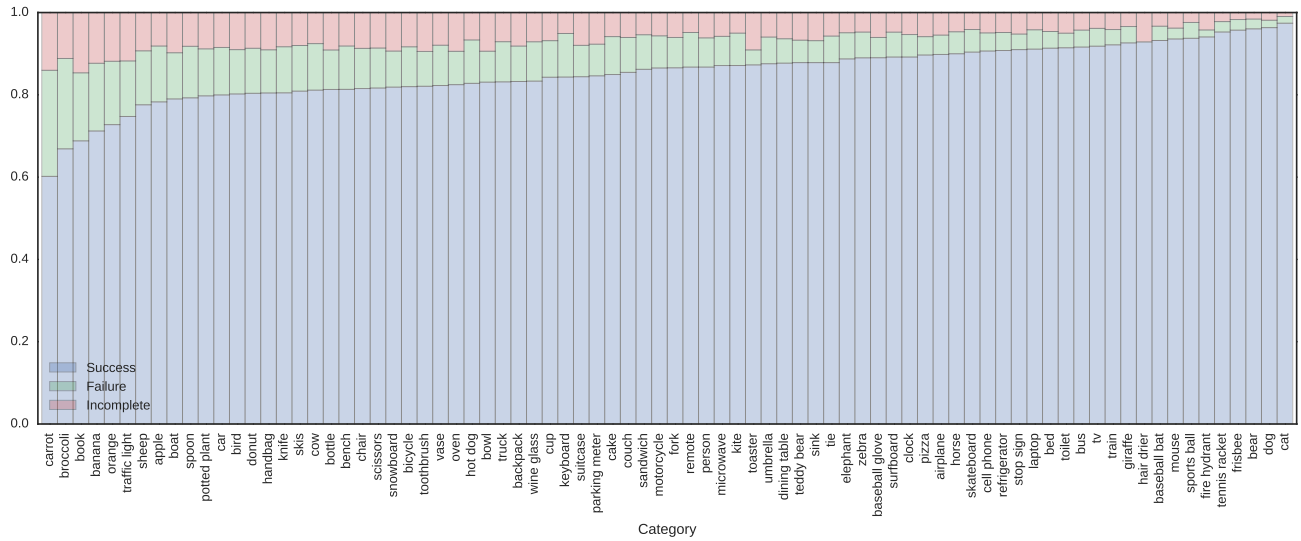
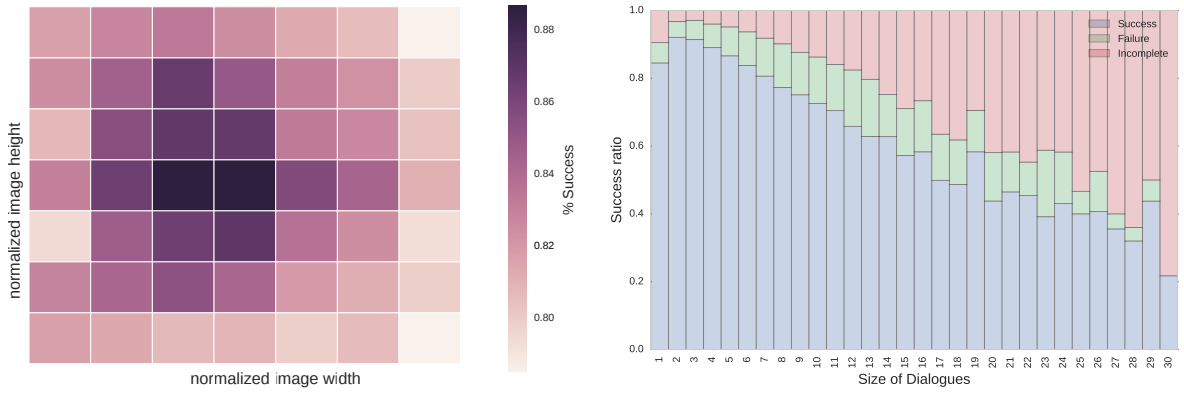


Figure 20: Histogram of success ratio broken down per object category.



(a) (b)

Figure 21: (a) Heatmap of the success ratio with respect to the spatial location within the picture. (b) Histogram of the success ratio relative to the dialogue length.

person	14.48	person	3.20	=	left	2.95	↗
food	1.29	left	1.69		right	2.32	↗
animal	1.16	wearing	1.20	new	person	2.28	↘
human	1.03	right	1.02	new	wearing	1.66	↘
object	0.77	front	0.97	new	whole	1.58	new
car	0.60	white	0.91	new	white	1.56	=
vehicle	0.57	red	0.77	new	red	1.26	=
cat	0.41	car	0.64	↘	black	1.19	↗
alive	0.37	black	0.60	new	front	1.14	↘
dog	0.35	blue	0.59	new	blue	1.10	=

(a) Dialogues having 3 questions

person	8.20	person	1.98	=	left	1.69	↗	left	1.92	=	left	2.13	=
food	1.03	left	1.03	↗	person	1.41	↘	right	1.77	↗	right	2.04	=
human	0.56	right	0.66	new	right	1.26	=	person	1.20	↘	white	1.28	↗
animal	0.46	front	0.59	new	white	0.84	↗	white	1.12	=	person	1.26	↘
vehicle	0.45	car	0.51	↗	wearing	0.82	↗	wearing	0.93	=	black	0.90	↗
object	0.42	white	0.48	new	side	0.67	↗	black	0.79	↗	wearing	0.85	↘
car	0.36	wearing	0.48	new	red	0.62	↗	red	0.72	=	red	0.80	=
furniture	0.24	side	0.43	new	front	0.58	↘	side	0.69	↘	whole	0.80	new
left	0.24	red	0.39	new	black	0.55	new	blue	0.65	↗	blue	0.75	=
edible	0.20	vehicle	0.39	↘	blue	0.54	new	front	0.58	↘	front	0.73	=

(b) Dialogues having 5 questions

person	5.89	person	1.44	=	left	1.08	↗	left	1.26	=	left	1.33	=	left	1.42	=	left	1.65	=
food	0.74	left	0.73	↗	person	0.96	↘	right	1.08	↗	right	1.22	=	right	1.39	=	right	1.54	=
human	0.38	right	0.42	new	right	0.89	=	person	0.82	↘	white	0.81	↗	white	0.88	=	white	0.96	=
vehicle	0.30	table	0.37	↗	side	0.57	↗	white	0.67	↗	person	0.80	↘	person	0.84	=	person	0.90	=
object	0.28	front	0.36	new	white	0.50	↗	side	0.60	↘	wearing	0.59	↗	black	0.63	↗	red	0.65	↗
car	0.26	food	0.35	↘	wearing	0.48	↗	wearing	0.54	=	side	0.57	↘	red	0.57	↗	black	0.63	↘
animal	0.26	side	0.35	new	red	0.41	new	red	0.49	=	red	0.54	=	wearing	0.56	↘	blue	0.57	↗
furniture	0.20	car	0.31	↘	table	0.39	↘	table	0.41	=	black	0.51	↗	blue	0.54	↗	wearing	0.52	↘
left	0.14	wearing	0.28	new	front	0.38	↘	black	0.41	↗	blue	0.49	↗	side	0.53	↘	next	0.51	new
boat	0.14	something	0.28	new	car	0.37	↘	blue	0.37	new	front	0.42	↗	front	0.45	=	side	0.51	↘

(c) Dialogues having 7 questions

Table 5: Proportions of the ten most common words for each depth of questions for sorted by the size of the dialogues

D. All oracle baselines

Model	Train err	Valid err	Test err
Dominant class (no)	47.4%	46.2%	50.9%
Category	43.0%	42.8%	43.1%
Question	40.2%	41.7%	41.2%
Crop	40.9%	42.7%	43.0%
Image	45.7%	46.7%	46.7%
Spatial	43.9%	44.1%	44.3%
Category + Spatial	41.6%	41.7%	42.1%
Question + Crop	22.3%	29.1%	29.2%
Question + Image	37.9%	40.2%	39.8%
Question + Category	23.1%	25.8%	25.7%
Question + Spatial	28.0%	31.2%	31.3%
Spatial + Crop	41.8%	42.4%	42.8%
Crop + Image	41.6%	42.1%	42.4%
Spatial + Image	42.2%	44.1%	44.2%
Category + Crop	41.0%	41.7%	42.3%
Category + Image	42.3%	42.7%	43.0%
Category + Crop + Image	40.6%	41.5%	41.8%
Category + Spatial + Crop	40.6%	41.6%	42.1%
Question + Category + Spatial	17.2%	21.1%	**21.5**%
Question + Crop + Image	23.7%	29.9%	30.0%
Category + Spatial + Image	40.4%	42.0%	42.2%
Question + Category + Image	23.4%	27.1%	27.4%
Question + Spatial + Image	28.4%	32.5%	32.5%
Spatial + Crop + Image	41.6%	42.1%	42.5%
Question + Category + Crop	20.4%	24.4%	24.7%
Question + Spatial + Crop	19.4%	26.0%	26.2%
Question + Category + Spatial + Crop	16.1%	21.7%	22.1%
Question + Spatial + Crop + Image	20.7%	27.7%	27.9%
Category + Spatial + Crop + Image	40.3%	41.4%	41.8%
Question + Category + Spatial + Image	19.2%	23.2%	23.5%
Question + Category + Crop + Image	20.0%	25.3%	25.5%
Question + Category + Spatial + Crop + Image	17.8%	23.2%	23.3%

Table 6: Classification errors for all oracle baselines.

E. Guesser generation model

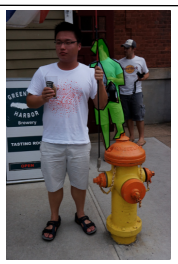
	Generated		Groundtruth	
	Is it a person?	No	Is it in the sky?	No
	Is it the kite?	No	Is it the umbrella?	No
	Is it the kite?	No	Is it the ocean?	No
	Is it the chair?	No	Is it the boat?	Yes
	Is it the boat?	Yes		
	Is it a person?	No	Is it an object?	Yes
	Is it a skateboard?	No	Do you wear it?	No
	Is it a car?	Yes	Do you ride it?	No
	Is it the one on the right?	No	Is it metal?	Yes
	Is it the one on the right?	No		
	Is it a person?	Yes	Is it a person?	Yes
	Is it the one in the front?	No	Is he in the foreground?	No
	Is it the one in the middle?	Yes	Is he wearing blue?	Yes
	Is it the one in the middle?	Yes		
	Is it the whole person?	Yes		

Figure 22: Three samples of QGen+GT model for which the correct object was predicted.

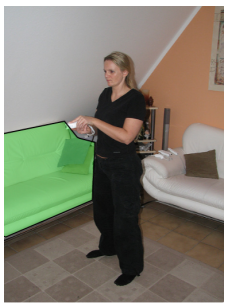
	Generated	Groundtruth
	Is it the cat? No Is it the cat? No Is it the chair? No Is it the book? No Is it the book? No	Is it an animal? No Is it a device? Yes Is it silver in color? Yes
	Is it a person? No Is it a remote? No Is it the chair? Yes Is it the one on the right? No Is it the one next to the right? No	Is it a person? No Is it a couch? Yes Does the couch have two pillows on it? Yes
	Is it a person? Yes Is it the guy in the front? No Is it the guy in the middle? No Is it the guy in the middle? No Is it the guy in the middle? No	Is it a person? Yes Is it in the foreground? No Is it on a screen? Yes

Figure 23: Three dialogue samples of QGen+GT model for which the wrong object was predicted.